COMP2211

University of Southampton

Software Engineering Group Project

Deliverable 2

Increment 1

Version History

No.	Date	Comments
1	16th March 2023	First Submission

Members

Name	Email
Gavin Teo Siu Chyi	gsct1c21@soton.ac.uk
Yohith V. Nyanasegaran	yvn1e20@soton.ac.uk
Lim Shi Xun	sxl1c21@soton.ac.uk
Ang Jing Ru	jra1c21@soton.ac.uk

Table of Contents

1 Response to Feedback

1.1 User Understanding

Feedback from User/Supervisor

During the deliverable review, user and supervisors brought up the question regarding citizen living near airport. They feel like this group of people should be included somewhere in the stakeholder analysis as they will be affected by the airport operations and we have not consider them in the stakeholder analysis.

Response to feedback

However, after further discussion, we have provided both the user and supervisors with the justification that the stakeholders should be those who will be using or affected by our system, in this case, a system to recalculate runway parameters when there is obstacles. Whilst the people is affected by noise or issues caused by airport operation, this is not being controlled by the system and so we will not be including them in the stakeholder list.

1.2 Requirements Planning

Feedback from User/Supervisor

Comments were given to the technical requirement section where the group have define reliability with criteria that system must perform without failure for 95% of the time. However, our user and supervisors think that 5% failure probability is still considered high for such an expert system.

Response to feedback

Instead of categorising and quantifying the criteria for each technical requirement, the updated technical requirements will be to have a system that has high availability so that it is available anytime anywhere. The system should also be reliable where calculation is carried out accurately without errors and perform with high efficiency. Appropriate error handling should also be implemented so that errors can be handled with ease and does not cause corruption of system or affect usage of the system.

1.3 Project Planning

Feedback from User/Supervisor

The sprint plan had been designed and shown to both user and supervisor prior the review, no further comments were given on this section.

Response to feedback

N/A

1.4 Project Set-up

Feedback from User/Supervisor

Some comments were given on the risk analysis. The user initially perceive mitigation as steps to be taken if the risks are realized. Upon discussion, user has agreed for the group to proceed with the different definition, which are ways to minimize the likelihood of the risk being realized. However, user has pointed out that the probability section should be altered as the probability of the risk will be reduced based on our definition of mitigation. He had suggested for us to keep track or to find a way to update the reducing of probability periodically so that the table will then reflect the actual risk exposure with the updated risk probability.

Response to feedback

Discussion will be held after each sprint so that we can update the risk analysis table with the correct probability after the mitigation steps were taken.

2 Application

2.1 Product Value

The Runway Redeclaration tool in this deliverable which gives real value by providing working code that helps users by providing aid for them in calculating the revised runway parameters when there is an obstacle on the runway. These runway parameter calculations and new revised values will provide guidance for the user to important key decisions regarding the airport runway operations to decide the landing and take-off operations of the runway.

The prototype produced up to this stage has a simple, easy to use and intuitive interface in which the user will be able to select a list of predefined runways and obstacles, then input the object's distance from threshold, which side of the runway the object is on, distance from centreline. The user can then click on the perform calculation button, and the tool will output (based on the flight method) the original and revised values of

Accelerate-Stop Distance Available (ASDA), Take-Off Distance Available (TODA), Take-Off Run Available (TORA) and Landing Distance Available (LDA).

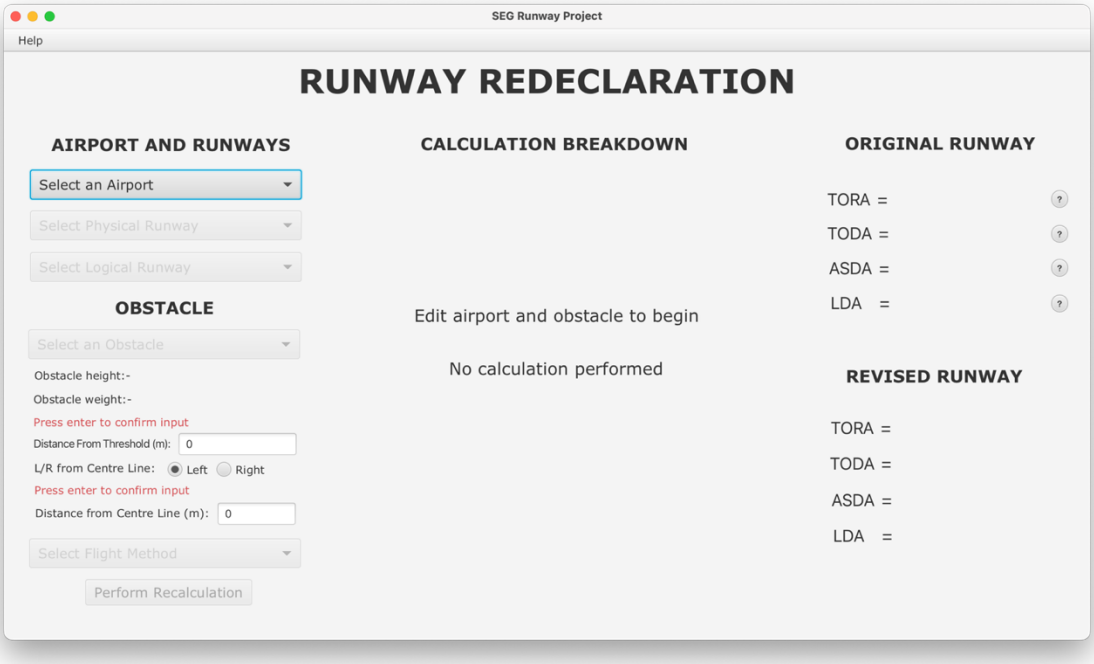
Even though the group has decided not to implement visualisation and customisation in this increment, the prototype itself is still considered important as it lays a strong foundation to the next two increments which will then rely heavily on the accuracy of the calculations.

2.2 Application Details

We have decided not to implement the edit and add functions for airport, runways and obstacles because it will involve updating and reading from XML files which will be very similar to what we plan to do in increment 3 (import and export details in XML files) and thus we will only be implementing it in the last increment. With this being said, we will be putting effort in ensuring the basic functionalities are working and leaving the customizing to the end.

2.2.1 Entry Page

Screenshot below show the entry page for our prototype in this increment, where there will be a leftmost pane for user input, and the rest of the area will be in the initial state without any calculation or details before user has keyed in the required details.

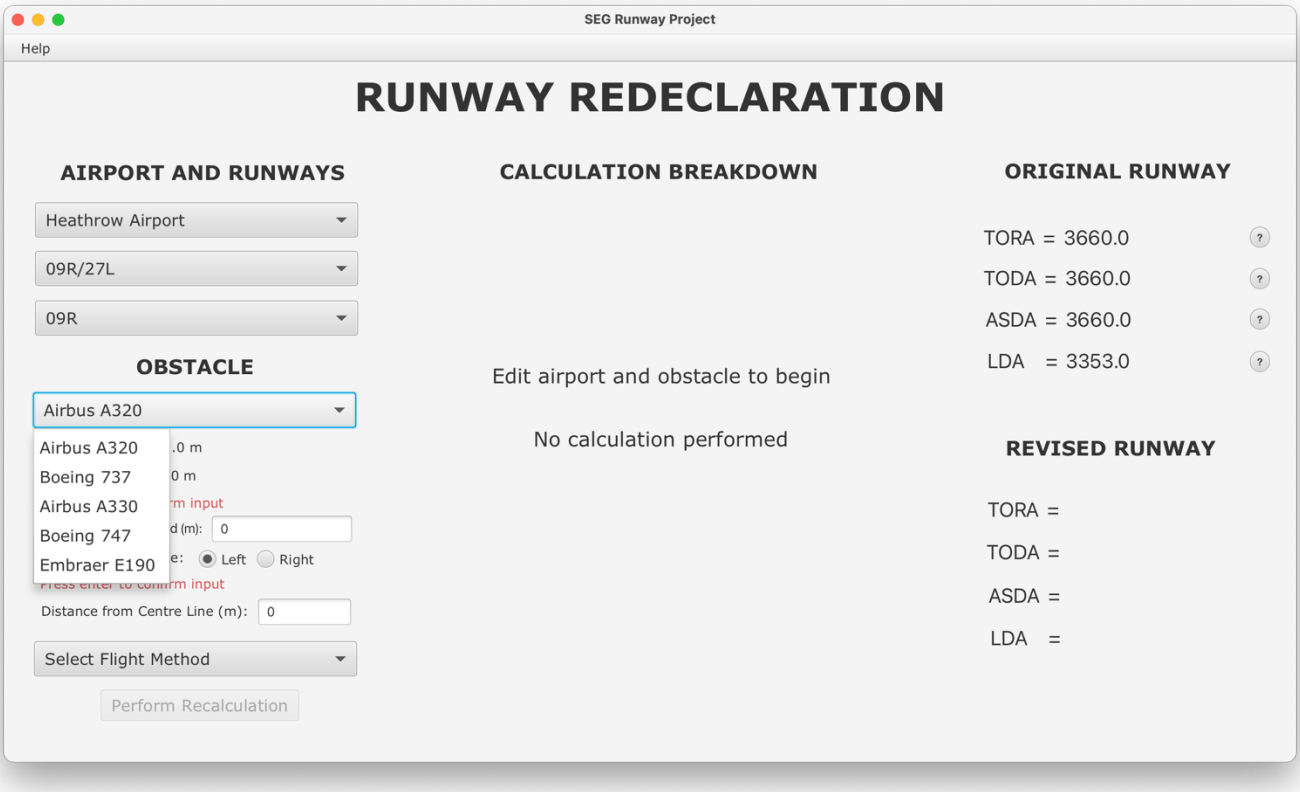


2.2.2 User Input and Predefined Data

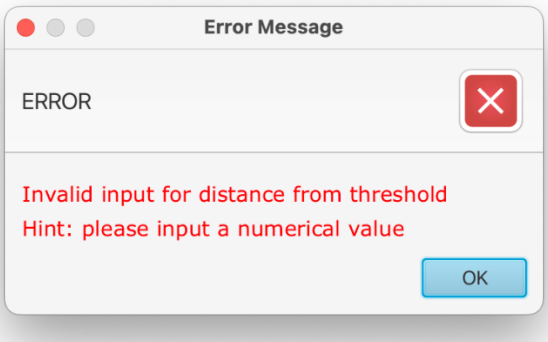
11	As an air traffic controller. I want to have a list of predefined obstacles. So that I can test out the program without creating a new obstacle.	Could

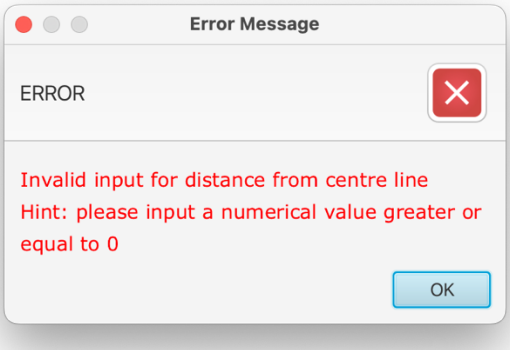
12	As an air traffic controller. I want the threshold that has the lowest value to always be on the left. So that it follows the formalized format.	Could

As seen from the screenshots below, the threshold is being represented in the format such that the lower value is located on the left. We have also defined a list of predefined obstacles to be used for testing, we have not implemented editing and adding new airports, runways and obstacles at this stage as we have planned to incorporated it in the last sprint.



If there are any error in input, for example, when user accidentally key in invalid numerical values for distance from threshold or distance from centreline, the error will be notified as below through a pop-up window and the text field will reset to a default value of 0.





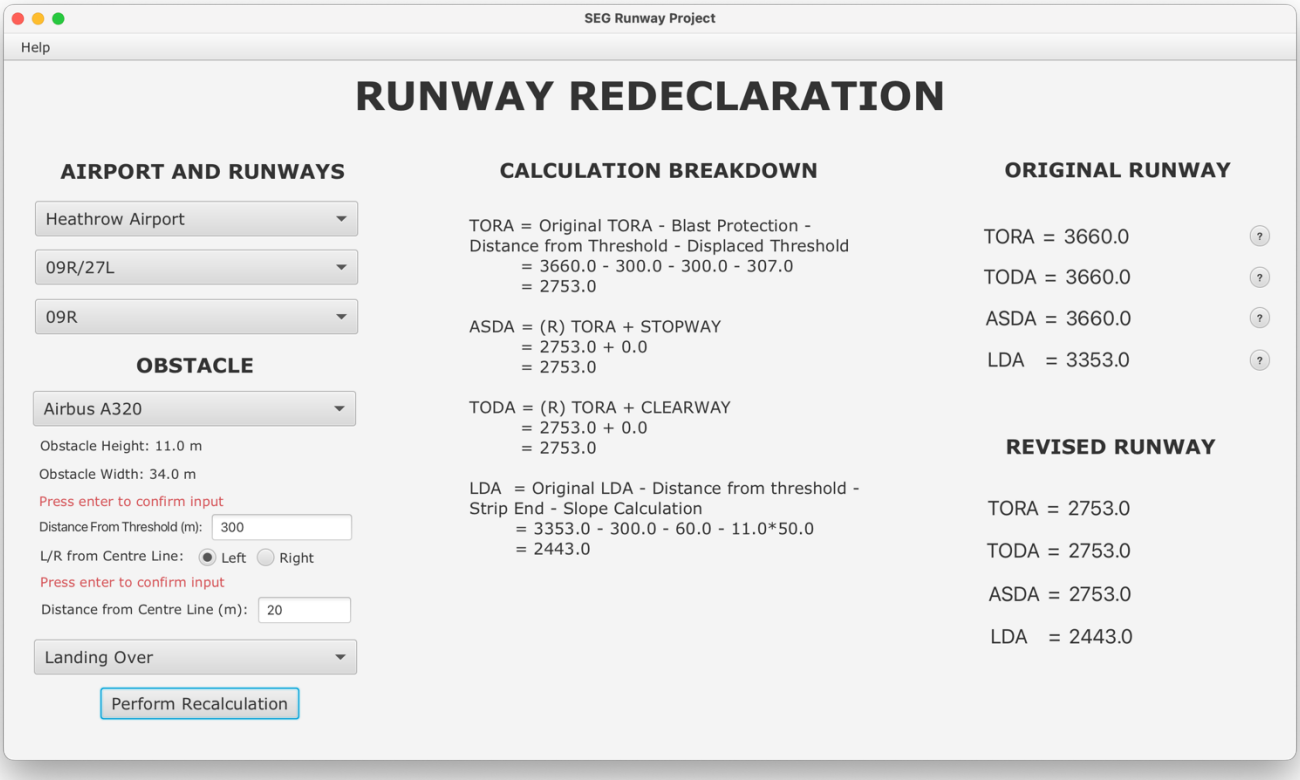
Below is the predefined obstacle xml file that will be loaded at the start of the application. These are some small UK aircrafts with their name, height and width extracted from the Internet.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<obstacles>
    <obstacle>
        <name>Airbus A320</name>
        <height>11</height>
        <width>34</width>
    </obstacle>
    <obstacle>
        <name>Boeing 737</name>
        <height>12</height>
        <width>35</width>
    </obstacle>
    <obstacle>
        <name>Airbus A330</name>
        <height>17</height>
        <width>60</width>
    </obstacle>
    <obstacle>
        <name>Boeing 747</name>
        <height>19</height>
        <width>69</width>
    </obstacle>
    <obstacle>
        <name>Embraer E190</name>
        <height>10</height>
        <width>28</width>
    </obstacle>
</obstacles>
```

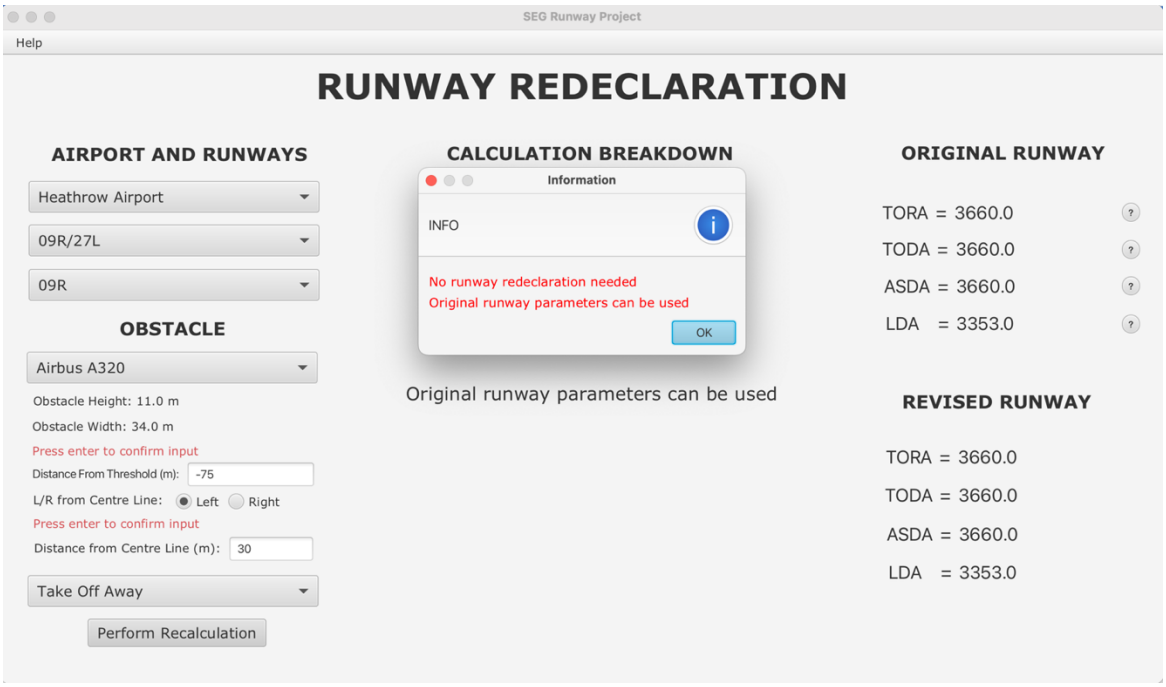
2.2.3 Calculation and Revising Runway Parameter

1	As an air traffic controller. I want to calculate the new runway distances when an obstacle is present. So that I can decide whether the official process is worthwhile	Must
2	As an air traffic controller. I want to be able to view the recalculated runway distances and the originals. So that the difference between them can be compared easily.	Must
6	As an air traffic controller. I want to be able to view the breakdown of the calculations. So that they can be compared with the paper results.	Should

As seen below, once user has selected the input for airport, runways and obstacle, they will be able to perform recalculation. The middle pane will display the calculation breakdown which can then be compared with manual calculation and the rightmost pane will show both the original and revised runway parameters so that it can be reviewed and compared easily.



If certain conditions are met such that no redeclaration is needed and the airplane can perform operation using the original runway parameter, the screen will be shown as below.



2.2.4 Additional Features (Help and Documentation)

In the toolbar, there is a 'Help' button, when clicked on, there will be an option of 'About runway project'. User will be directed to project definition for this runway project.

There are a column of '?' button on the right side of all the runway parameters, when mouse is hovered on, a text box with description and explanation for each parameters will be shown so that it is easier for user to understand what each parameters is representing.

3 Design

3.1 Design Choices

3.1.1 Model-View-Controller Architecture

Model-View-Controller or MVC is a design pattern where the program is comprised of three components. The model, which represents the data logic and interacts with the database so it's responsible for all validating, updating, deleting and saving of data. For example, the model would be obstacles, airport, physical and logical runways and each of their parameters, flight method and revised logical runway. The next one is the view, it's the user interface of the program and reflects data taken from the model. In this case the model will be the dashboard for selection and input as well as the displays for calculation. Lastly is the controller, it acts as a middleman between the previous components, it receives input from the view, sends it to the model and receives back the new data which is updated in the view. The controller will be the runway display, error handling and XML handling.

3.2 Design Artifacts

In the first increment, we implemented Unified Modelling Language (UML), a modelling tool that was used for planning and designing the general structure of our application. We decided to include use case diagram, sequence diagram, class diagram, and storyboard as key design Artifact along side with a user scenario.

3.2.1 User Scenarios

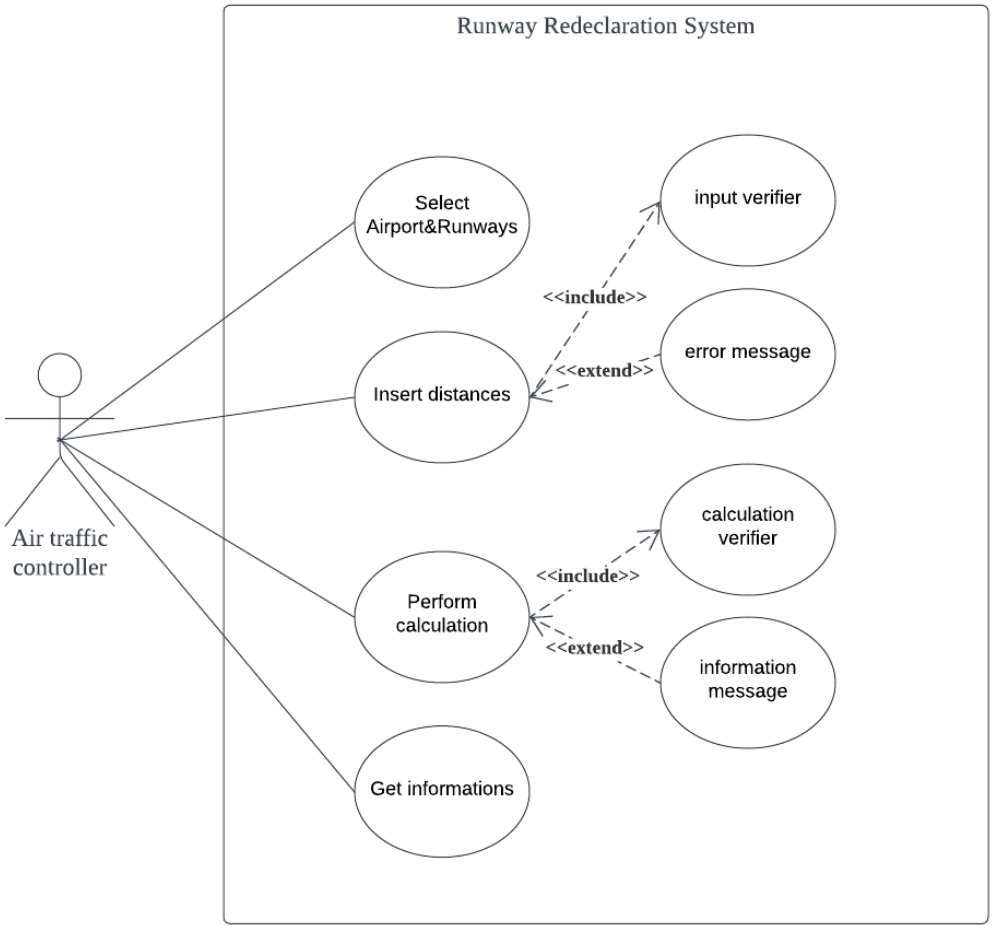
User scenarios were introduced during the development phase so that our team would have a better understanding on how users interact with the system.

3.2.1.1 Scenario 1

In scenario 1, the user (air traffic controller) will try to perform runway redeclaration calculations based on a list of predefined airports, physical runways, logical runways, and obstacles. After the calculation was performed, the user will be able to compare the original values with the revised values and compare the calculation breakdown with their paperwork.

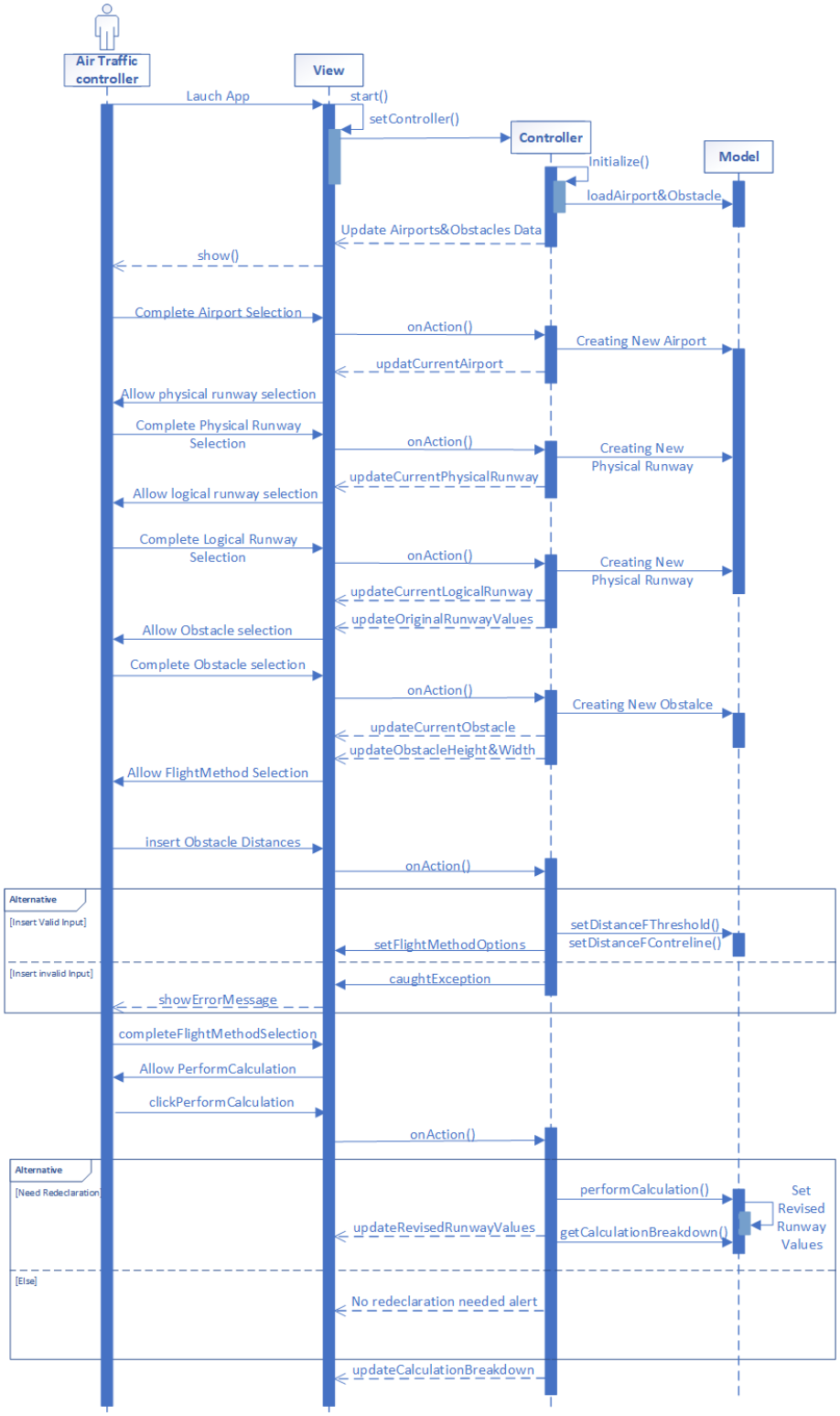
3.2.2 Use Case Diagram

This use case diagram was built based on the scenario above. The main focus of this diagram is to visualise how air traffic controllers interact with the system.



3.2.3 Sequence Diagram

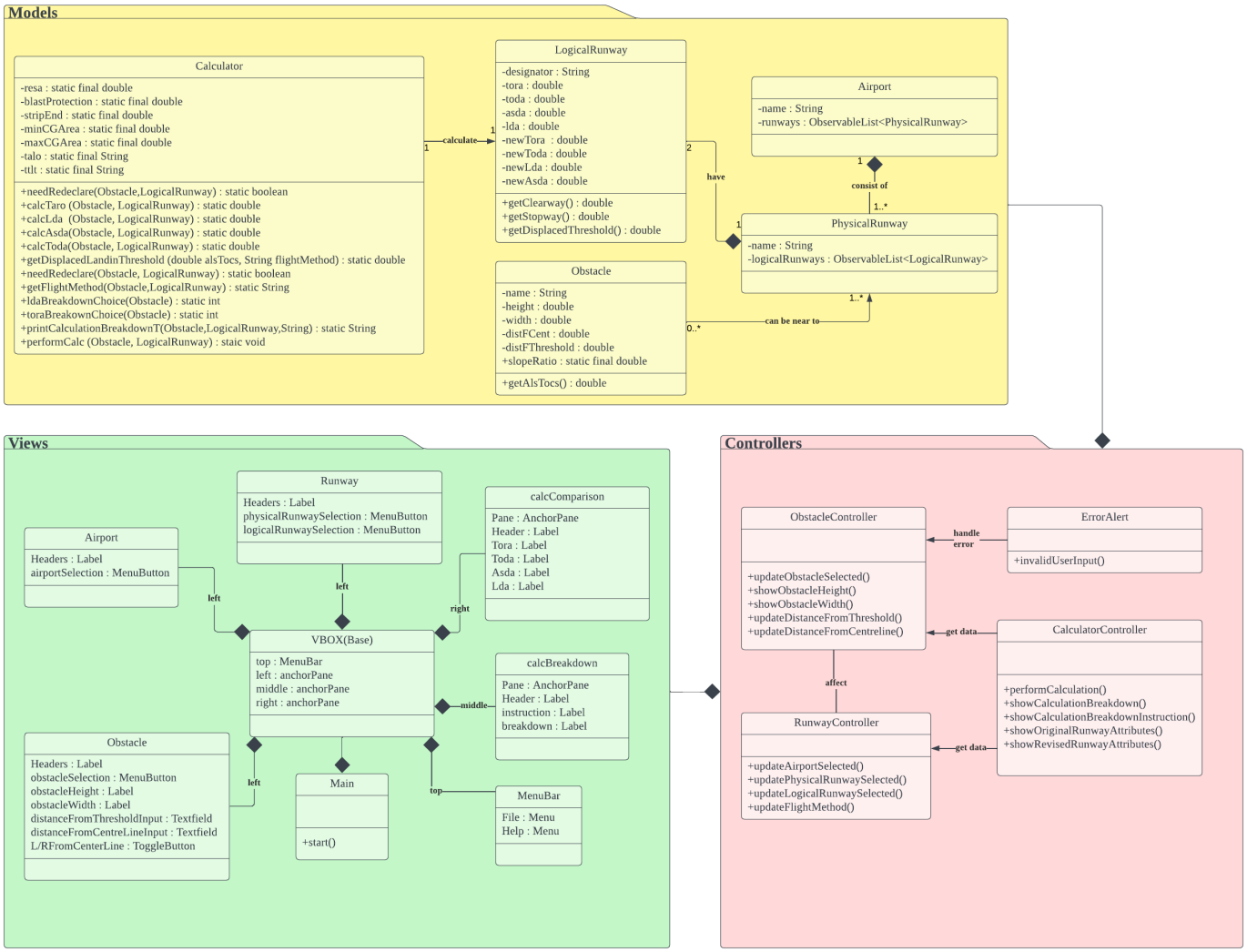
This sequence diagram illustrates how the actor (air traffic controller) interacts with a different part of the system that was based on MVC architecture. It also shows how systems interact with each other when a task is performed by the user.



3.2.4 Class Diagram

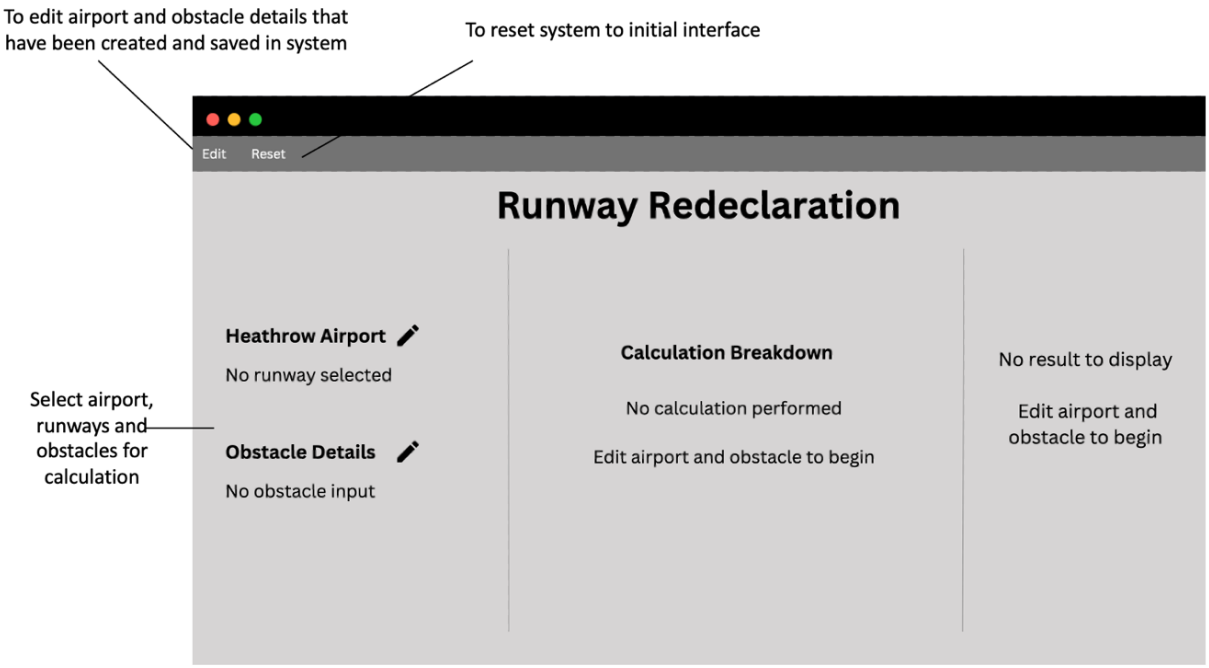
A class diagram was created to provide a clear understanding of the system structure among all the team members. This allows us to further divide our tasks in more detail while also working towards the same goal even when we are not physically working together all the time.

We are using the Model View Controller framework (MVC) to design the application. This class diagram consists of 3 different packages, which are, Models, Views, and Controllers. Each of them represents a different part of the system which has its functionalities respectively.

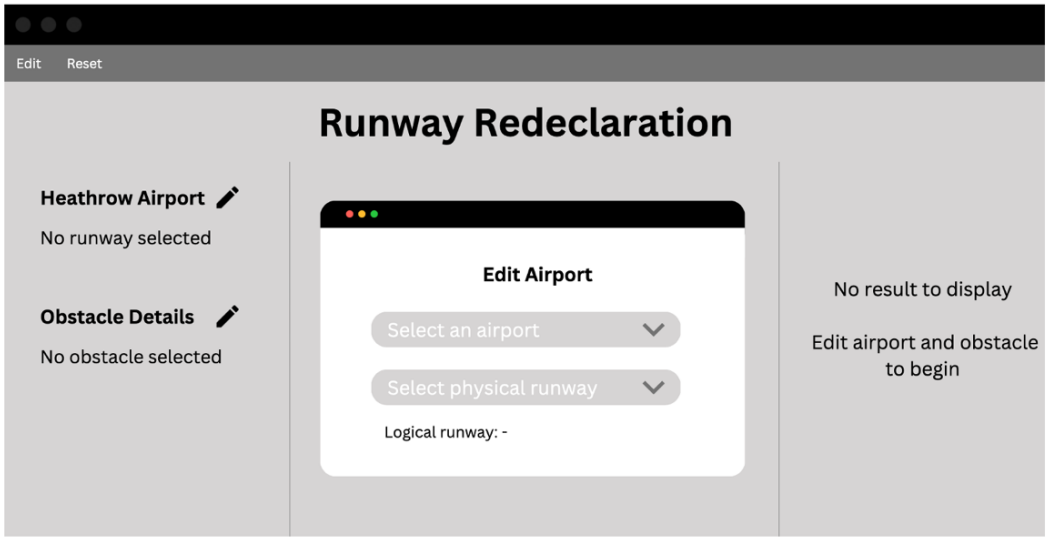


3.2.5 Storyboards

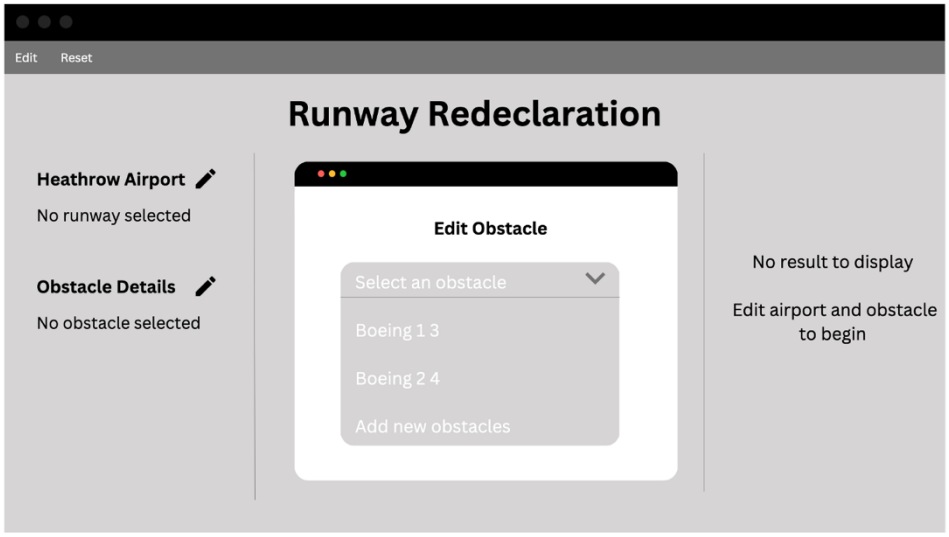
Initial version of storyboards created for the runway declaration system which includes four pages displaying the essential interfaces we intend to incorporate. The interfaces are currently displayed in monochrome, and we will determine the colour scheme later in the project. Our goal is to opt for a minimalist design that suits professional use. Our priority for this project will be the system's functionality and comprehensiveness than producing an aesthetically pleasing app.



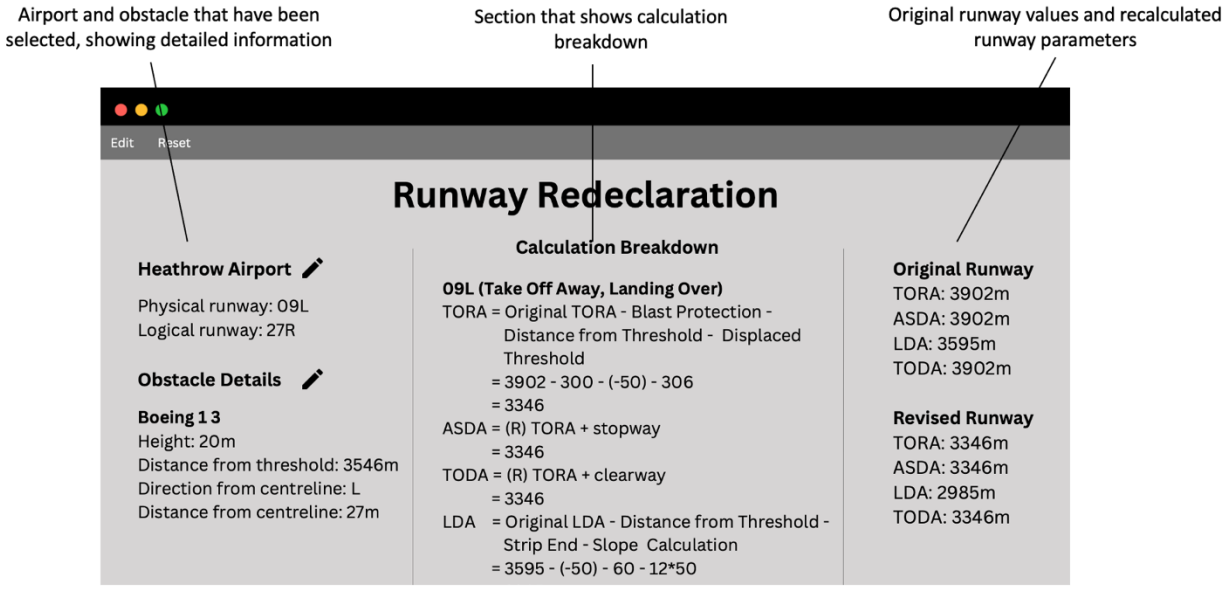
1. Starting page (Pressing reset from other page will direct user to this page)



2. Pop-up window when user click on edit button beside airport name



3. Pop up window when user clicks on edit button next to obstacle details



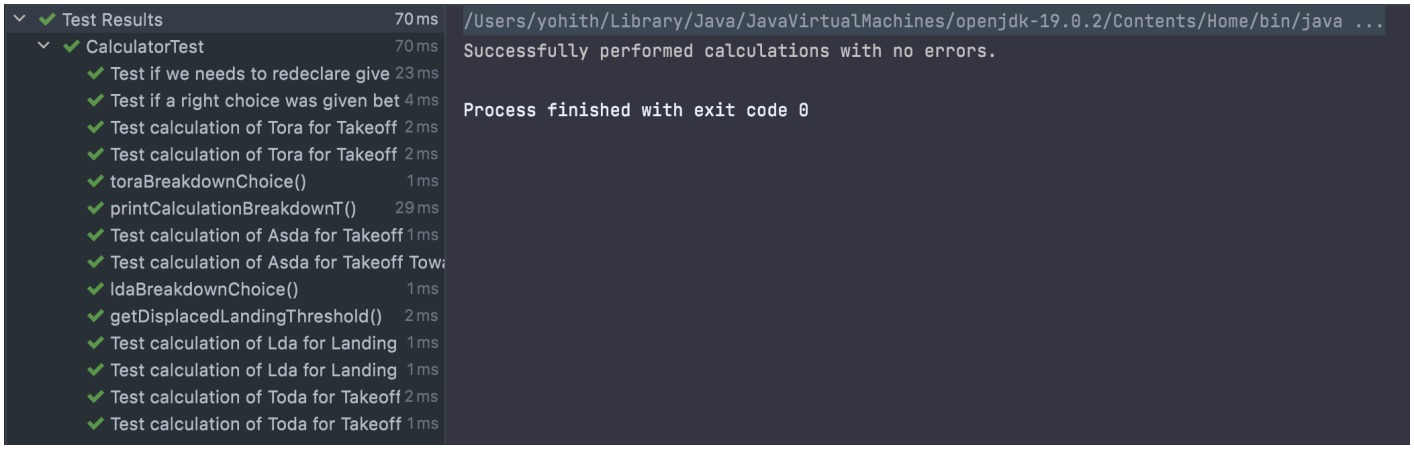
4. Calculation interface where calculation breakdown and result are shown

4 Product Testing

4.1 Test results

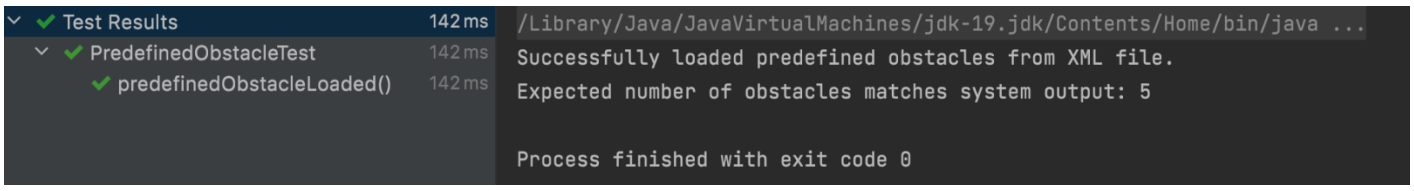
4.1.1 Test for Calculations

Test was performed to check if the system is able to perform calculations accurately and with no errors. Screenshot below shows the test output.



4.1.2 Test for Predefined Obstacles

Test was conducted to check if the system is able to load the full predefined list of obstacles from xml file without exception. Screenshot below shows the test output.



5 First Sprint Review

The following paragraphs are the key points of our sprint review. It consists of the successes we had during this sprint as well as challenges, which the team will seek to improve during the following sprint.

5.1 Successes and Challenges

Successes

- Efficient planning/delegating is the most significant strength of the group. During each sprint, a different leader would be assigned, this allows each of the team members to have the opportunity to lead while giving the previous leader the chance to rest, this sprint is Shi Xun. During the first deliverable, when it was roughly 90% complete, a plan for the second deliverable has already been in place. Right after the first deliverable was submitted, the team was able to immediately start working on the second deliverable and thus able to complete it earlier than planned. This gave the team sufficient time for testing as well as consultation with the user/supervisor.
- Regular meetings were scheduled, which allowed the team members to update/communicate their progress with each other thus clearing up any miscommunications and ensuring the sprint is not stagnant.

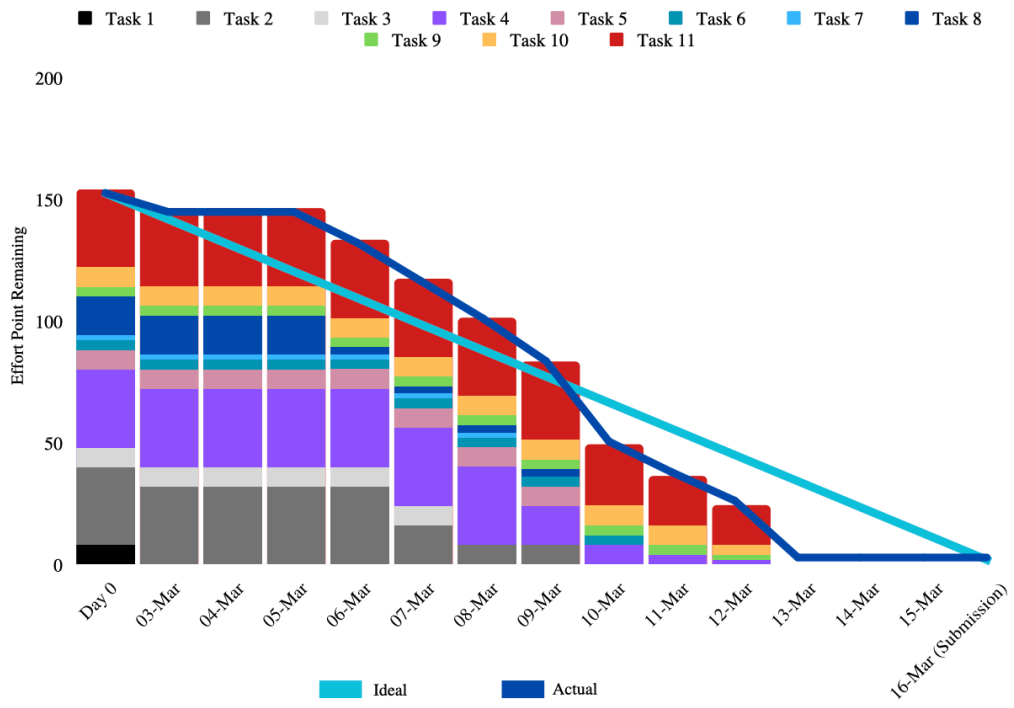
Challenges and Solutions

As great as the team's planning were, it still has flaws. One key part of the sprint was report writing but the team was too fixated on the coding segment that the writing segment was left unassigned and neglected. The task was eventually assigned but only after when the code was mostly complete. In the next increment, the group has included tasks for report writing along with the main coding tasks to ensure both are being worked on equally.

5.2 Sprint Burn-down Chart

Table 1.1 Tasks for First Sprint: List of tasks for first sprint

No.	Tasks	Effort (pts)
	Application - Functionalities	
1.	Research and setting up GitHub for project	8
2.	Code for calculation of new runway distances	32
3.	Have a list of obstacles to be used for testing	8
4.	A basic interface design and layout using Scene Builder	32
5.	Show originals and calculated values	8
6.	Show calculation breakdown	4
7.	Threshold displayed based on format, with lower on the left	2
	Design and Planning	
8.	Using UML diagrams, storyboards and scenarios to support design decisions	16
9.	Construction of burndown chart for increment 1 based on actual progress	4
10.	Introduction of sprint plan for increment 2	8
	Testing	
11.	Test and verifying the correctness of application	32



6 Second Increment Planning

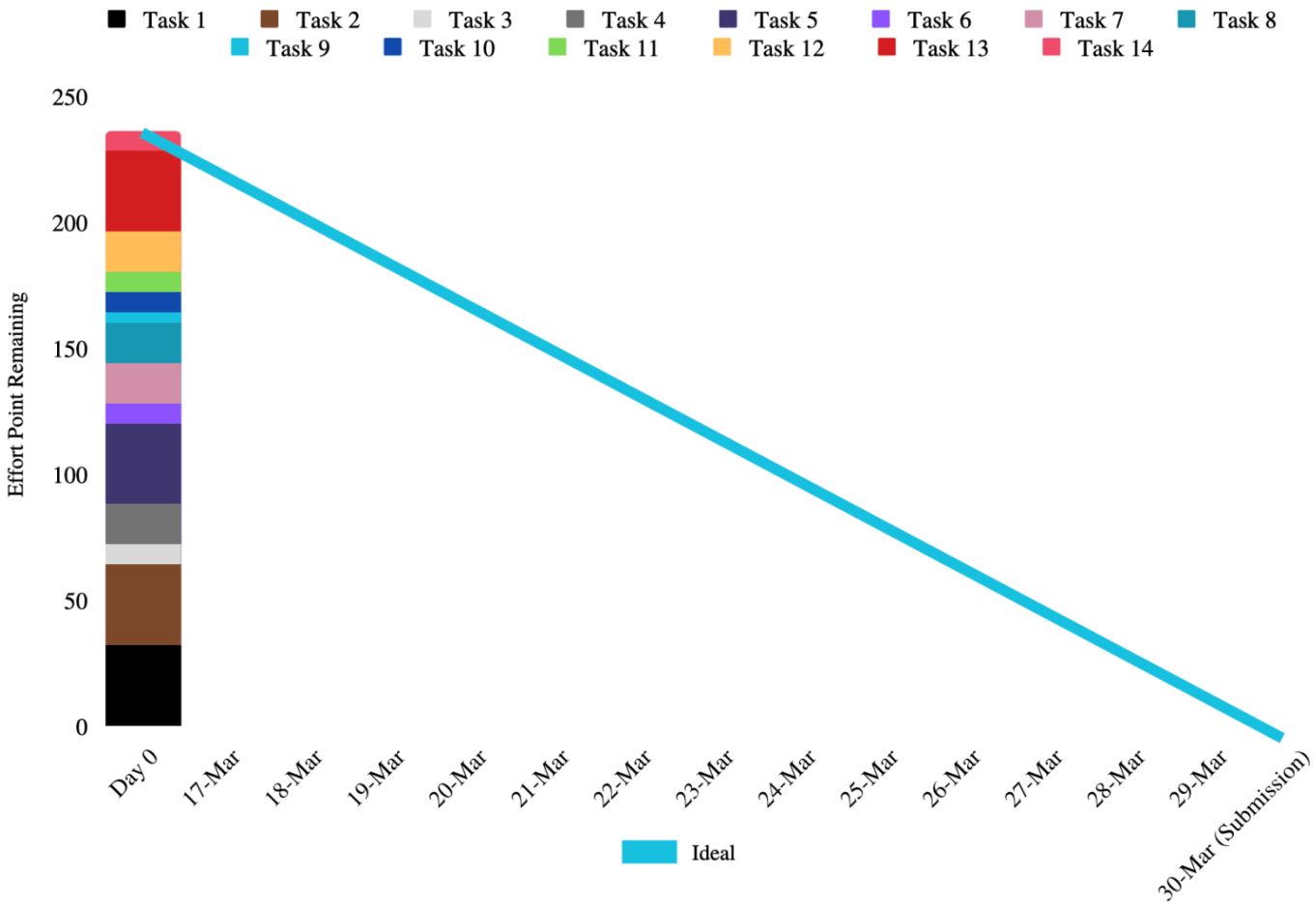
6.1 Sprint Plan for the Next Increment

As an air traffic controller. I want to view the airport from top-down and side-on views simultaneously or individually in 2D. So that I can better visualize the runway.
As an air traffic controller. I want to view the cleared and graded areas in top-down view. So that I can manage the area more effectively.
As an air traffic controller. I want to view the details of the runway and the distances used for runway declaration from both views. So that I can have a better understanding of how the runway declaration was done in the system.
As an air traffic controller. I want to select different runways and thresholds, with views changing accordingly. So that the application is compatible with different runways.
As an air traffic controller. I want to view the TOCS, and ALS slope caused by the obstacles in side-on view. So that I can make a better-informed decision.
As an air traffic controller, I want the threshold that has the lowest value to always be on the left. So that it follows the formalized format.
As an air traffic controller. I want to have notifications displayed for any actions that occur. So that I am fully aware of what has taken place.

Table 1.2 Tasks for Second Sprint: List of tasks for second sprint with effort points determined using planning poker

No.	Tasks	Effort (pts)
	Application - Functionalities	
1.	Create a basic layout for side view	32
2.	Create a basic layout for top-down view	32
3.	Clear display for cleared and graded area	8
4.	Clear representation of new runway parameters in the visualizations	16
5.	Views changing dynamically based on runways selected and other inputs	32
6.	TOCS and ALS slope caused by obstacles being shown clearly in side-view	8
7.	Notification feature to keep track of changes to the system	16
	Planning and Report Writing	
8.	Using UML diagrams, storyboards to support design decisions	16
9.	Construction of burndown chart for second increment based on actual progress	4
10.	Introduction of sprint plan for third increment	8
11.	Response to feedback	8
12.	Application demo and screenshot	16
	Testing	
13.	Test and verifying the correctness of application	32
	Submission	
14.	Finalizing submission folder, exporting .jar file, writing readme	8

6.2 Day-Zero Burndown Chart



7 Additional Feedbacks

During the meeting with supervisor and demonstration of the prototype, user suggested the possibility of inputting obstacle details by taking photo and estimating height and width based on the picture instead of inputting it by measuring and typing it manually.

Upon discussion with supervisor, he thinks that this idea is feasible with research done to find out to understand the software needed and how to implement it. However, given the time frame given for each increment, he feels that this idea is not feasible within the time limit.